

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION FOR UNITED STATES PATENT

FOR

**SYSTEM, DEVICE, AND METHOD FOR SENDING KEEP-ALIVE MESSAGES
IN A COMMUNICATION NETWORK**

Inventor:

Bradley Cain
295 Harvard Street #804
Cambridge, MA 02139

Attorney Docket: 2204/186 (BA414)

Attorneys:

BROMBERG & SUNSTEIN LLP
125 Summer Street
Boston, MA 02110
(617) 443-9292

09457209-120399
USPTO-60275460

-1-

SYSTEM, DEVICE, AND METHOD FOR SENDING KEEP-ALIVE MESSAGES IN A COMMUNICATION NETWORK

5

FIELD OF THE INVENTION

The present invention relates generally to communication systems, and more particularly to sending keep-alive messages in a communication system.

10

BACKGROUND OF THE INVENTION

In today's information age, computers and computer peripherals are often internetworked over a communication network. The communication network includes a number of network nodes that interoperate to route protocol messages within the communication network. These network nodes typically run various routing protocols in order to determine forwarding paths for routing protocol messages within the communication network.

15

20

When a network node fails, the other network nodes need to route the protocol messages around the failed network node. The network nodes typically rely on "keep-alive" messages to determine whether a particular network node is operational. Each node periodically sends keep-alive messages to its neighbors. A network node may consider a particular neighbor to be operational as long as the neighbor is sending keep-alive messages.

25

Therefore, each network node receives keep-alive messages from its neighbors. The processing of keep-alive messages can be computationally intensive, especially if the network node has many neighbors.

30

SUMMARY OF THE INVENTION

In accordance with one aspect of the invention, the frequency for sending keep-alive messages to a neighbor is determined based upon a reliability factor for communicating with the neighbor. A node determines a reliability factor for

communicating with a neighbor and sets the frequency for sending keep-alive messages to the neighbor based upon the reliability factor. The reliability factor is determined based upon the reliability of the neighbor as well as the reliability of the communication link to the neighbor. The frequency for sending keep-alive messages to the neighbor is relatively high if the reliability factor is low. The frequency for sending keep-alive messages to the neighbor is relatively low if the reliability factor is high. The frequency for sending keep-alive messages to the neighbor is dynamically adjusted based upon an updated reliability factor.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing and other objects and advantages of the invention will be appreciated more fully from the following further description thereof with reference to the accompanying drawings wherein:

FIG. 1 is a logic flow diagram showing exemplary logic for determining a frequency for sending keep-alive messages to a neighbor in accordance with an embodiment of the invention;

FIG. 2 is a logic flow diagram showing exemplary logic for determining a reliability factor in accordance with an embodiment of the invention;

FIG. 3 is a logic flow diagram showing exemplary logic for adjusting the frequency for sending keep-alive messages to a neighbor in accordance with an embodiment of the invention; and

FIG. 4 is a logic flow diagram showing exemplary logic for adjusting frequency for sending keep-alive messages to a neighbor based upon an updated reliability factor in accordance with an embodiment of the invention.

DETAILED DESCRIPTION OF A PREFERRED EMBODIMENT

An exemplary embodiment of the present invention sets the frequency for sending keep-alive messages to a particular neighbor based upon a reliability factor for

communicating with the neighbor. The keep-alive messages are sent at a relatively low frequency if the reliability factor for communicating with the neighbor is high. This is because there is a relatively high likelihood that each keep-alive message will be received and processed by the neighbor, so fewer keep-alive messages are needed to keep the communication link to the neighbor active. The keep-alive messages are sent at a relatively high frequency if the reliability factor for communicating with the neighbor is low. This is because there is a relatively low likelihood that each keep-alive message will be received and processed by the neighbor, so more keep-alive messages are needed to keep the communication link to the neighbor active. The reliability factor is updated regularly, and the frequency for sending keep-alive message to the neighbor is dynamically adjusted accordingly.

Specifically, each node determines a reliability factor for communicating with each of its neighbors, and sets the frequency of keep-alive messages for each neighbor based upon the corresponding reliability factor. The reliability factor is preferably based upon the reliability of the neighbor as well as the reliability of the communication link to the neighbor, both of which can be determined to some degree and can vary over time. The node continually updates the reliability factor and adjusts the frequency of keep-alive messages accordingly.

More particularly, the node determines a reliability for the neighbor and a reliability for the communication link to the neighbor using predetermined reliability determination techniques. For example, the node may determine the reliability of the communication link to the neighbor by measuring communication link characteristics, such as signal-to-noise ratio, harmonic distortion, phase hits, gain hits, carrier-to-noise ratio, bit error rate, block error rate, packet error rate, cell loss ratio, congestion, or other characteristics. The node may determine the reliability for the neighbor based upon keep-alive or other messages received from the neighbor, including explicit status information provided by the neighbor. However, the present invention is in no way limited to any particular technique or techniques for determining the reliability of the neighbor and/or the reliability of the communication link to the neighbor.

Once the node determines the reliability for the neighbor and the reliability for the

communication link to the neighbor, the node determines the reliability factor based upon the reliability for the neighbor and the reliability for the communication link to the neighbor. In an exemplary embodiment of the invention, the node assigns a relative weight to each reliability determination. The reliability factor is the weighted average of the two reliability determinations. Thus the reliability factor RF can be represented by the following equation:

Eq. 1
$$RF = (W1 * A + W2 * B)$$

where A is the reliability for the communication link to the neighbor, B is the reliability for the neighbor, W1 is the relative weight for A, and W2 is the relative weight for B.

The weights W1 and W2 are implementation-specific. In an exemplary weighting scheme, the weights W1 and W2 are initially set equal. The weights may be adjusted thereafter as network conditions change.

After determining the reliability factor RF for communicating with the neighbor, the node determines the frequency for sending keep-alive messages to the neighbor based upon the reliability factor RF. The frequency for sending keep-alive messages to the neighbor is inversely proportional to the reliability factor RF. Thus, the frequency for sending keep-alive messages to the neighbor is relatively low if the reliability factor RF is high, and the frequency for sending keep-alive messages to the neighbor is relatively high if the reliability factor RF is low.

FIG. 1 is a logic flow diagram showing exemplary logic 100 for determining a frequency for sending keep-alive messages to a neighbor. Beginning at step 102, the logic first determines a reliability factor for communicating with a neighbor, in step 104. The logic then determines a frequency for sending keep-alive messages to the neighbor based upon the reliability factor, in step 106. The logic 100 terminates in step 199.

FIG. 2 is a logic flow diagram showing exemplary logic 200 for determining the reliability factor. Beginning at step 202, the logic determines a reliability for the neighbor, in step 204, and also determines a reliability for the communication link to the neighbor, in step 206. The logic then assigns a relative weight to each of the reliability for the neighbor

and the reliability for the communication link to the neighbor, in step 208. The logic then determines the reliability factor to be the weighted average of the reliability for the neighbor and the reliability for the communication link to the neighbor, in step 210. The logic then sets the frequency for sending keep-alive messages to the neighbor in inverse proportion to the reliability factor, in step 212. The logic 200 terminates in step 299.

FIG. 3 is a logic flow diagram showing exemplary logic 300 for adjusting the frequency for sending keep-alive messages to a neighbor. Beginning at step 302, the logic proceeds to determine an updated reliability factor, in step 304. The logic then adjusts the frequency for sending keep-alive messages to the neighbor based upon the updated reliability factor, in step 306. The logic 300 terminates in step 399.

FIG. 4 is a logic flow diagram showing exemplary logic 400 for adjusting frequency for sending keep-alive messages to a neighbor based upon an updated reliability factor. Beginning at step 402, the logic determines whether the updated reliability factor represents a reliability improvement or a reliability degradation, in step 404. The logic increases the frequency for sending keep-alive messages to the neighbor if the updated reliability factor represents a reliability degradation, in step 406. The logic decreases the frequency for sending keep-alive messages to the neighbor if the updated reliability factor represents a reliability improvement, in step 408. The logic 400 terminates in step 499.

In the described embodiments, the reliability factor is proportional to the reliability for communicating with the neighbor, and therefore the frequency for sending keep-alive messages to the neighbor is inversely proportional to the reliability factor. However, in an alternative embodiment of the invention, the reliability factor may be inversely proportional to the reliability for communicating with the neighbor (for example, the reliability factor may be an error measurement that increases as the reliability decreases), in which case the frequency for sending keep-alive messages to the neighbor is proportional to the reliability factor. The present invention is in no way limited by the technique for determining the reliability factor nor by the technique for deriving the frequency for sending keep-alive messages based upon the reliability factor.

In an exemplary embodiment of the present invention, predominantly all of the logic for sending keep-alive messages described herein is implemented as a set of

computer program instructions that are stored in a computer readable medium and executed by an embedded microprocessor system within a network node. Various embodiments of the invention may be implemented in any conventional computer programming language. For example, an embodiment may be implemented in a procedural programming language (*e.g.*, "C") or an object oriented programming language (*e.g.*, "C++"). Alternative embodiments of the invention may be implemented using discrete components, integrated circuitry, programmable logic used in conjunction with a programmable logic device such as a Field Programmable Gate Array (FPGA) or microprocessor, or any other means including any combination thereof.

Alternative embodiments of the invention may be implemented as a computer program product for use with a computer system. Such implementation may include a series of computer instructions fixed either on a tangible medium, such as a computer readable media (*e.g.*, a diskette, CD-ROM, ROM, or fixed disk), or fixed in a computer data signal embodied in a carrier wave that is transmittable to a computer system via a modem or other interface device, such as a communications adapter connected to a network over a medium. The medium may be either a tangible medium (*e.g.*, optical or analog communications lines) or a medium implemented with wireless techniques (*e.g.*, microwave, infrared or other transmission techniques). The series of computer instructions embodies all or part of the functionality previously described herein with respect to the system. Those skilled in the art should appreciate that such computer instructions can be written in a number of programming languages for use with many computer architectures or operating systems. Furthermore, such instructions may be stored in any memory device, such as semiconductor, magnetic, optical or other memory devices, and may be transmitted using any communications technology, such as optical, infrared, microwave, or other transmission technologies. It is expected that such a computer program product may be distributed as a removable medium with accompanying printed or electronic documentation (*e.g.*, shrink wrapped software), preloaded with a computer system (*e.g.*, on system ROM or fixed disk), or distributed from a server or electronic bulletin board over the network (*e.g.*, the Internet or World Wide Web).

The present invention may be embodied in other specific forms without departing

-7-

from the essence or essential characteristics. The described embodiments are to be considered in all respects only as illustrative and not restrictive.

0945/209-12039
66037-603/5460